UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/749,615	12/30/2003	Gregor K. Frey	6570P031	8289

45062   7590   07/07/2009
SAP/BSTZ
BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP
1279 OAKMEAD PARKWAY
SUNNYVALE, CA 94085-4040

EXAMINER
LINDSEY, MATTHEW S

ART UNIT	PAPER NUMBER
2451	

MAIL DATE	DELIVERY MODE
07/07/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

PTOL-90A (Rev. 04/07)

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *19 March 2009*.

2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *31-48* is/are pending in the application.

　　4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *31-48* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

　　Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

　　Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

　　a)☐ All   b)☐ Some * c)☐ None of:

　　　1.☐ Certified copies of the priority documents have been received.

　　　2.☐ Certified copies of the priority documents have been received in Application No. _____.

　　　3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

　* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO/SB/08)
　　Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
　　Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application

6)☐ Other: _____.

DETAILED ACTION

1.   Claims 31-48 are pending in this application. Claims 1-30 are canceled and

Claims 31-48 are new as filed on 19 March 2009.

Claim Rejections - 35 USC § 112

2.   The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of
> making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the
> art to which it pertains, or with which it is most nearly connected, to make and use the same and shall
> set forth the best mode contemplated by the inventor of carrying out his invention.

3.   Claims 31, 36, 37, 42, 43 and 48 are rejected under 35 U.S.C. 112, first

paragraph, as failing to comply with the written description requirement. The claim(s)

contains subject matter which was not described in the specification in such a way as to

reasonably convey to one skilled in the relevant art that the inventor(s), at the time the

application was filed, had possession of the claimed invention.

4.   With respect to Claims 31, 36, 37, 42, 43 and 48, they recite the limitation:

"informing the logging controller that the logging controller is to write received messages

into both the log file and the trace file so that the tracing controller is no longer used to

write received messages into the trace file", where the logging controller is used to write

both log messages and trace messages to the log file and trace file and informing the

logging controller to perform such. Examiner has searched the original disclosure and found only that the logging controller or trace controller can write to both the console and to a file (see for example, Specification, [0023]), however this is not the same as the logging controller writing to both the log file and trace file such that the trace controller is no longer used to write received messages into the trace file.

Claim Rejections - 35 USC § 103

5.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6.   **Claims 31, 37 and 43 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shah (US 2003/0005173) in view of Ullman (US 7,120,685) and further in view of Sluiman (US 2004/0237093).**

7.   With respect to Claim 31, Shah disclosed: "A method, comprising: selecting a severity level for a controller ([0050], lines 1-8, and [0054], lines 5-7, where the severity level is the configuration of the filter, for example the amount of data passed to handlers from the filter can change based on the filter configuration and thus the severity level is selected)",

"instantiating a logging controller from the controller ([0043], lines 1-4, where a

handler is a logging controller associated with a message logger, see [0042], lines 2-5),

the logging controller inheriting the severity level from the controller ([0059], lines

3-9),

the logging controller to receive logging messages from various categories of

software within an enterprise information system ([0042], lines 4-5),

the logging messages having varied levels of severity ([0050], lines 1-8, where

messages are filtered according to type, or severity),

the logging controller designed to write into a log file those of the received

logging messages having a severity level that is higher than a first maximum severity

level setting ([0050], lines 4-8, where a filter determines what types of messages are

sent to a destination such as a file, see [0045] and therefore the filter has a set

maximum severity level setting and only messages having a higher severity level are

sent to the file),

the logging controller designed to not write into the log file those of the received

logging messages having a severity level that is lower than a first minimum severity

level setting ([0050], lines 4-8, where a filter determines what types of messages are

sent to a destination such as a file, see [0045], and therefore the filter has a set

minimum severity level setting and messages having a lower severity level are not sent

to the file),

the inherited severity level being below said first maximum severity level setting

and above said first minimum severity level setting ([0059], lines 3-9, where it is

conceivable that the parent has a severity level setting below the maximum severity

level and above the minimum severity level settings);

       instantiating a tracing controller from the controller ([0043], lines 1-4, where a

handler is a tracing controller associated with a trace logger, see [0042], lines 6-9),

       the tracing controller inheriting the severity level from the controller ([0059], lines

3-9),

       the tracing controller to receive tracing messages from various software locations

within the enterprise information system ([0042], lines 9-11),

       the tracing messages having varied levels of severity ([0050], lines 1-8, where

messages are filtered according to type, or severity),

       the tracing controller designed to write into a trace file those of the received

tracing messages having a severity level that is higher than a second maximum severity

level setting ([0050], lines 4-8, where a filter determines what types of messages are

sent to a destination such as a file, see [0045] and therefore the filter has a set

maximum severity level setting and only messages having a higher severity level are

sent to the file),

       the tracing controller designed to not write into the trace file those of the received

tracing messages having a severity level that is lower than a second minimum severity

level setting ([0050], lines 4-8, where a filter determines what types of messages are

sent to a destination such as a file, see [0045], and therefore the filter has a set

minimum severity level setting and messages having a lower severity level are not sent

to the file),

the inherited severity level being below said second maximum severity level

setting and above said second minimum severity level setting ([0059], lines 3-9, where it

is conceivable that the parent has a severity level setting below the maximum severity

level and above the minimum severity level settings);

setting the first and second maximum and minimum security levels of the

respective logging and tracing controllers equal to the inherited severity level ([0052],

lines 5-8, where if the message logger, trace logger and their respective handlers and

filters are part of a group, in the situation where the severity level of the handler

changes, the filters will inherit that level and thus the first and second maximum and

minimum inherited severity levels of the filters will be set to the inherited severity level of

the handlers) to configure the logging and tracing controllers to write into their

respective log and trace files messages whose severity level is above the inherited

severity level and not write into their respective log and trace files messages whose

severity level is below the inherited severity level ([0050], lines 4-8);

subsequent to said logging and tracing controllers writing received messages into

their respective log and trace files ([0045] and Fig 5)",

"informing the logging controller that the logging controller is to write received

messages into both the log file and the trace file so that the tracing controller is no

longer used to write received tracing messages into the trace file ([0043], lines 1-5 and

Fig. 7, where multiple loggers can send their output to one handler, in Fig 7 message

logger 703, or logging controller, and trace logger 704, or tracing controller, both send

output to file handler 706 which is in contrast to Fig 5, where trace logger and message

logger send their output to different handlers)".

Shah did not explicitly state: "the severity level being selected from a plurality of

available severity levels that at least include: a) a first severity level that indicates the

existence of an anomaly that an application can recover from, said application also able

to perform a desired task notwithstanding the anomaly; b) a second severity level that

indicates the existence of an error that an application can recover from, said application

also being unable to perform a desired task because of the error; said selecting of said

severity level including selecting one of a), b) above", or

"correlating the specific locations of software within the enterprise information

system to the specific categories of software within the enterprise information system".

However, Ullman disclosed: "the severity level being selected from a plurality of

available severity levels (Col. 3, lines 50-60) wherein, available severity levels include:

a) a first severity level that indicates the existence of an anomaly that an

application can recover from, said application also able to perform a desired task (Col.

3, lines 53-60, specifically warning messages, e.g., 'The available RAM is dangerously

low' indicates the existence of an anomaly that an application can recover from and still

perform a desired task);

b) a second severity level that indicates the existence of an error that an

application can recover from (Col. 1, lines 29-33, where an example error is a

misspelling on the input, and applications can recover from user misspelling on input),

said application also being unable to perform a desired task (Col. 3, lines 53-60,

specifically error messages, e.g., 'An error has occurred in this program' indicates the

desired task had an error);

said selecting of said severity level including selecting one of a), b) above (Col. 4,

lines 24-29)".

It would have been obvious to one of ordinary skill in the art at the time of the

invention to modify the logging system of Shah with the teachings of Ullman to include

support for variable levels of logging. Motivation to combine these references comes

from saving disk space by allowing the logger to only record events that are needed for

diagnostics. Therefore by combining the references one can save performance while

still logging relevant events.

The combination of Shah and Ullman did not explicitly state: "correlating the

specific locations of software within the enterprise information system to the specific

categories of software within the enterprise information system".

However, Sluiman disclosed: "correlating the specific locations of software within

the enterprise information system to the specific categories of software within the

enterprise information system ([0004], lines 1-8, where correlation can occur between

trace records and log records)".

It would have been obvious to one of ordinary skill in the art at the time of the

invention to modify the logging system of Shah and Ullman with the teachings of

Sluiman to include support for correlating message log categories and trace log

locations. Motivation to combine these references comes from Sluiman, where:

"correlation is used to determine relationships (both implicit and explicit) between

instrumentation information captured in instrumentation artefacts generated by an

instrumentation service" ([0004], lines 3-6). Therefore, by combining the references,

one can correlate the events in the log to determine relationships and therefore better

understand the events.

8.   With respect to Claim 37, Shah disclosed: "An article of manufacture comprising

program code stored on a computer readable storage medium, said program code to be

read from said computer readable storage medium and processed by one or more

processors to perform a method ([0029]), comprising:",

        "instantiating a logging controller from the controller ([0043], lines 1-4, where a

handler is a logging controller associated with a message logger, see [0042], lines 2-5),

        the logging controller inheriting the severity level from the controller ([0059], lines

3-9),

        the logging controller to receive logging messages from various categories of

software within an enterprise information system ([0042], lines 4-5),

        the logging messages having varied levels of severity ([0050], lines 1-8, where

messages are filtered according to type, or severity),

        the logging controller designed to write into a log file those of the received

logging messages having a severity level that is higher than a first maximum severity

level setting ([0050], lines 4-8, where a filter determines what types of messages are sent to a destination such as a file, see [0045] and therefore the filter has a set maximum severity level setting and only messages having a higher severity level are sent to the file),

the logging controller designed to not write into the log file those of the received logging messages having a severity level that is lower than a first minimum severity level setting ([0050], lines 4-8, where a filter determines what types of messages are sent to a destination such as a file, see [0045], and therefore the filter has a set minimum severity level setting and messages having a lower severity level are not sent to the file),

the inherited severity level being below said first maximum severity level setting and above said first minimum severity level setting ([0059], lines 3-9, where it is conceivable that the parent has a severity level setting below the maximum severity level and above the minimum severity level settings);

instantiating a tracing controller from the controller ([0043], lines 1-4, where a handler is a tracing controller associated with a trace logger, see [0042], lines 6-9),

the tracing controller inheriting the severity level from the controller ([0059], lines 3-9),

the tracing controller to receive tracing messages from various software locations within the enterprise information system ([0042], lines 9-11),

the tracing messages having varied levels of severity ([0050], lines 1-8, where messages are filtered according to type, or severity),

the tracing controller designed to write into a trace file those of the received

tracing messages having a severity level that is higher than a second maximum severity

level setting ([0050], lines 4-8, where a filter determines what types of messages are

sent to a destination such as a file, see [0045] and therefore the filter has a set

maximum severity level setting and only messages having a higher severity level are

sent to the file),

the tracing controller designed to not write into the trace file those of the received

tracing messages having a severity level that is lower than a second minimum severity

level setting ([0050], lines 4-8, where a filter determines what types of messages are

sent to a destination such as a file, see [0045], and therefore the filter has a set

minimum severity level setting and messages having a lower severity level are not sent

to the file),

the inherited severity level being below said second maximum severity level

setting and above said second minimum severity level setting ([0059], lines 3-9, where it

is conceivable that the parent has a severity level setting below the maximum severity

level and above the minimum severity level settings);

setting the first and second maximum and minimum security levels of the

respective logging and tracing controllers equal to the inherited severity level ([0052],

lines 5-8, where if the message logger, trace logger and their respective handlers and

filters are part of a group, in the situation where the severity level of the handler

changes, the filters will inherit that level and thus the first and second maximum and

minimum inherited severity levels of the filters will be set to the inherited severity level of

the handlers) to configure the logging and tracing controllers to write into their respective log and trace files messages whose severity level is above the inherited severity level and not write into their respective log and trace files messages whose severity level is below the inherited severity level ([0050], lines 4-8);

subsequent to said logging and tracing controllers writing received messages into their respective log and trace files ([0045] and Fig 5)",

"informing the logging controller that the logging controller is to write received messages into both the log file and the trace file so that the tracing controller is no longer used to write received tracing messages into the trace file ([0043], lines 1-5 and Fig. 7, where multiple loggers can send their output to one handler, in Fig 7 message logger 703, or logging controller, and trace logger 704, or tracing controller, both send output to file handler 706 which is in contrast to Fig 5, where trace logger and message logger send their output to different handlers)".

Shah did not explicitly state: "the severity level being selected from a plurality of available severity levels that at least include: a) a first severity level that indicates the existence of an anomaly that an application can recover from, said application also able to perform a desired task notwithstanding the anomaly; b) a second severity level that indicates the existence of an error that an application can recover from, said application also being unable to perform a desired task because of the error; said selecting of said severity level including selecting one of a), b) above", or

"correlating the specific locations of software within the enterprise information

system to the specific categories of software within the enterprise information system".

However, Ullman disclosed: "the severity level being selected from a plurality of

available severity levels (Col. 3, lines 50-60) wherein, available severity levels include:

a) a first severity level that indicates the existence of an anomaly that an

application can recover from, said application also able to perform a desired task (Col.

3, lines 53-60, specifically warning messages, e.g., 'The available RAM is dangerously

low' indicates the existence of an anomaly that an application can recover from and still

perform a desired task);

b) a second severity level that indicates the existence of an error that an

application can recover from (Col. 1, lines 29-33, where an example error is a

misspelling on the input, and applications can recover from user misspelling on input),

said application also being unable to perform a desired task (Col. 3, lines 53-60,

specifically error messages, e.g., 'An error has occurred in this program' indicates the

desired task had an error);

said selecting of said severity level including selecting one of a), b) above (Col. 4,

lines 24-29)".

It would have been obvious to one of ordinary skill in the art at the time of the

invention to modify the logging system of Shah with the teachings of Ullman to include

support for variable levels of logging. Motivation to combine these references comes

from saving disk space by allowing the logger to only record events that are needed for

diagnostics. Therefore by combining the references one can save performance while still logging relevant events.

The combination of Shah and Ullman did not explicitly state: "correlating the specific locations of software within the enterprise information system to the specific categories of software within the enterprise information system".

However, Sluiman disclosed: "correlating the specific locations of software within the enterprise information system to the specific categories of software within the enterprise information system ([0004], lines 1-8, where correlation can occur between trace records and log records)".

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the logging system of Shah and Ullman with the teachings of Sluiman to include support for correlating message log categories and trace log locations. Motivation to combine these references comes from Sluiman, where: "correlation is used to determine relationships (both implicit and explicit) between instrumentation information captured in instrumentation artefacts generated by an instrumentation service" ([0004], lines 3-6). Therefore, by combining the references, one can correlate the events in the log to determine relationships and therefore better understand the events.

9.   With respect to Claim 43, Shah disclosed: "A computer comprising program code stored in memory of said computer system, said memory coupled to one or more

processors of said computing system, said program code to be read from said memory

and processed by said one or more processors to perform a method ([0029]),

comprising:",

"instantiating a logging controller from the controller ([0043], lines 1-4, where a

handler is a logging controller associated with a message logger, see [0042], lines 2-5),

the logging controller inheriting the severity level from the controller ([0059], lines

3-9),

the logging controller to receive logging messages from various categories of

software within an enterprise information system ([0042], lines 4-5),

the logging messages having varied levels of severity ([0050], lines 1-8, where

messages are filtered according to type, or severity),

the logging controller designed to write into a log file those of the received

logging messages having a severity level that is higher than a first maximum severity

level setting ([0050], lines 4-8, where a filter determines what types of messages are

sent to a destination such as a file, see [0045] and therefore the filter has a set

maximum severity level setting and only messages having a higher severity level are

sent to the file),

the logging controller designed to not write into the log file those of the received

logging messages having a severity level that is lower than a first minimum severity

level setting ([0050], lines 4-8, where a filter determines what types of messages are

sent to a destination such as a file, see [0045], and therefore the filter has a set

minimum severity level setting and messages having a lower severity level are not sent

to the file),

the inherited severity level being below said first maximum severity level setting

and above said first minimum severity level setting ([0059], lines 3-9, where it is

conceivable that the parent has a severity level setting below the maximum severity

level and above the minimum severity level settings);

instantiating a tracing controller from the controller ([0043], lines 1-4, where a

handler is a tracing controller associated with a trace logger, see [0042], lines 6-9),

the tracing controller inheriting the severity level from the controller ([0059], lines

3-9),

the tracing controller to receive tracing messages from various software locations

within the enterprise information system ([0042], lines 9-11),

the tracing messages having varied levels of severity ([0050], lines 1-8, where

messages are filtered according to type, or severity),

the tracing controller designed to write into a trace file those of the received

tracing messages having a severity level that is higher than a second maximum severity

level setting ([0050], lines 4-8, where a filter determines what types of messages are

sent to a destination such as a file, see [0045] and therefore the filter has a set

maximum severity level setting and only messages having a higher severity level are

sent to the file),

the tracing controller designed to not write into the trace file those of the received

tracing messages having a severity level that is lower than a second minimum severity

level setting ([0050], lines 4-8, where a filter determines what types of messages are

sent to a destination such as a file, see [0045], and therefore the filter has a set

minimum severity level setting and messages having a lower severity level are not sent

to the file),

the inherited severity level being below said second maximum severity level

setting and above said second minimum severity level setting ([0059], lines 3-9, where it

is conceivable that the parent has a severity level setting below the maximum severity

level and above the minimum severity level settings);

setting the first and second maximum and minimum security levels of the

respective logging and tracing controllers equal to the inherited severity level ([0052],

lines 5-8, where if the message logger, trace logger and their respective handlers and

filters are part of a group, in the situation where the severity level of the handler

changes, the filters will inherit that level and thus the first and second maximum and

minimum inherited severity levels of the filters will be set to the inherited severity level of

the handlers) to configure the logging and tracing controllers to write into their

respective log and trace files messages whose severity level is above the inherited

severity level and not write into their respective log and trace files messages whose

severity level is below the inherited severity level ([0050], lines 4-8);

subsequent to said logging and tracing controllers writing received messages into

their respective log and trace files ([0045] and Fig 5)",

"informing the logging controller that the logging controller is to write received

messages into both the log file and the trace file so that the tracing controller is no

longer used to write received tracing messages into the trace file ([0043], lines 1-5 and

Fig. 7, where multiple loggers can send their output to one handler, in Fig 7 message

logger 703, or logging controller, and trace logger 704, or tracing controller, both send

output to file handler 706 which is in contrast to Fig 5, where trace logger and message

logger send their output to different handlers)".

      Shah did not explicitly state: "the severity level being selected from a plurality of

available severity levels that at least include: a) a first severity level that indicates the

existence of an anomaly that an application can recover from, said application also able

to perform a desired task notwithstanding the anomaly; b) a second severity level that

indicates the existence of an error that an application can recover from, said application

also being unable to perform a desired task because of the error; said selecting of said

severity level including selecting one of a), b) above", or

      "correlating the specific locations of software within the enterprise information

system to the specific categories of software within the enterprise information system".

      However, Ullman disclosed: "the severity level being selected from a plurality of

available severity levels (Col. 3, lines 50-60) wherein, available severity levels include:

      a) a first severity level that indicates the existence of an anomaly that an

application can recover from, said application also able to perform a desired task (Col.

3, lines 53-60, specifically warning messages, e.g., 'The available RAM is dangerously

low' indicates the existence of an anomaly that an application can recover from and still

perform a desired task);

b) a second severity level that indicates the existence of an error that an

application can recover from (Col. 1, lines 29-33, where an example error is a

misspelling on the input, and applications can recover from user misspelling on input),

said application also being unable to perform a desired task (Col. 3, lines 53-60,

specifically error messages, e.g., 'An error has occurred in this program' indicates the

desired task had an error);

said selecting of said severity level including selecting one of a), b) above (Col. 4,

lines 24-29)".

It would have been obvious to one of ordinary skill in the art at the time of the

invention to modify the logging system of Shah with the teachings of Ullman to include

support for variable levels of logging. Motivation to combine these references comes

from saving disk space by allowing the logger to only record events that are needed for

diagnostics. Therefore by combining the references one can save performance while

still logging relevant events.

The combination of Shah and Ullman did not explicitly state: "correlating the

specific locations of software within the enterprise information system to the specific

categories of software within the enterprise information system".

However, Sluiman disclosed: "correlating the specific locations of software within

the enterprise information system to the specific categories of software within the

enterprise information system ([0004], lines 1-8, where correlation can occur between trace records and log records)".

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the logging system of Shah and Ullman with the teachings of Sluiman to include support for correlating message log categories and trace log locations. Motivation to combine these references comes from Sluiman, where: "correlation is used to determine relationships (both implicit and explicit) between instrumentation information captured in instrumentation artefacts generated by an instrumentation service" ([0004], lines 3-6). Therefore, by combining the references, one can correlate the events in the log to determine relationships and therefore better understand the events.

10.   Claims 32-36, 38-42 and 44-48 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shah Ullman and Sluiman in view of Log file logging levels (August 14, 2003).

11.   With respect to Claims 32, 38 and 44, the combination of Shah, Ullman and Sluiman disclosed: "wherein the available security levels further include: a fifth severity level whose messages are an echo of what has been performed (Ullman, Col. 3, lines 60-61)".

The motivation to combine is the same as that mentioned above in Claim 31.

The combination of Shah, Ullman and Sluiman did not explicitly state: "a third severity level whose corresponding messages contain information for debugging; a fourth severity level that permits its corresponding messages to contain path information for looping and branching".

However, Log file logging levels disclosed: "a third severity level whose corresponding messages contain information for debugging (Table 4-1, Debug level); a fourth severity level that permits its corresponding messages to contain path information for looping and branching (Table 4-1, Entry_Exit level)".

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the logging system of Shah Ullman and Sluiman with the teachings of Log file logging levels to include support for variable levels of logging. Motivation to combine these references comes from saving disk space by allowing the logger to only record events that are needed for diagnostics. Therefore by combining the references one can save performance while still logging relevant events.

12.   With respect to Claims 33, 39 and 45, the combination of Shah, Ullman, Sluiman and Log file logging levels disclosed: "wherein the first and second severity levels are higher severity than the third, fourth and fifth severity levels (Log file logging levels, Table 4-1, where the error and warning messages are of higher severity than the debug, entry_exit and everything levels)".

The motivation to combine is the same as that mentioned above in Claim 32.

13.   With respect to Claims 34, 40 and 46, the combination of Shah, Ullman, Sluiman

and Log file logging levels disclosed: " wherein respective severity levels for at least a

portion of the received logging and tracing messages are determined by assigning the

respective severity levels to the respective software within the enterprise system that

generated the portion of the received logging and tracing messages (Shah, [0050], lines

1-3, where filters can be applied to loggers, which are software objects that record

events, see [0042], lines 1-2)".

14.   With respect to Claims 35, 41 and 47, the combination of Shah, Ullman, Sluiman

and Log file logging levels disclosed: "wherein the correlating of the specific locations of

software within the enterprise information system to specific categories of software

within the enterprise information system is submitted through a category application

programming interface (API) (Sluiman, [0017], lines 1-11, where API calls are made to

request correlators for transactions)".

The motivation to combine is the same as that mentioned above in Claim 31.

15.   With respect to Claims 36, 42 and 48, the combination of Shah, Ullman, Sluiman

and Log file logging levels disclosed: "wherein the informing of the logging controller

that the logging controller is to write received messages into both the log file and the

trace file is conducted through the messages received by the logging controller that are

written into both the log file and the trace file (Shah, Fig. 7 and [0043], lines 1-5, where

a file handler, 706, receives output from message logger 703 and trace logger 704, and outputs data to local file 708, which serves as the log file and the trace file)".

Response to Arguments

16.   Applicant's arguments with respect to claims 31-48 have been considered but are moot in view of the new ground(s) of rejection.

17.   Applicant's arguments filed 19 March 2009, see pg 13 have been fully considered but they are not persuasive.

18.   Applicant argues: "neither of these references appear to disclose setting first and second maximum and minimum severity levels of respective logging and tracing controllers equal to an inherited severity level to configure the logging and tracing controllers to write into their respective log and trace files messages whose severity level is above the inherited severity level and not write into their respective log and trace files messages whose severity level is below the inherited severity level" (pg 13, lines 3-9, not including cited portion).

Examiner respectfully disagrees. Shah disclosed: "setting the first and second maximum and minimum security levels of the respective logging and tracing controllers equal to the inherited severity level ([0052], lines 5-8, where if the message logger,

trace logger and their respective handlers and filters are part of a group, in the situation

where the severity level of the handler changes, the filters will inherit that level and thus

the first and second maximum and minimum inherited severity levels of the filters will be

set to the inherited severity level of the handlers) to configure the logging and tracing

controllers to write into their respective log and trace files messages whose severity

level is above the inherited severity level and not write into their respective log and trace

files messages whose severity level is below the inherited severity level ([0050], lines 4-

8)".

19.   Applicant further argues: "The Shah and Ullman references also appear to fail to

disclose informing a logging controller that the logging controller is to write received

messages into both the log and the trace file so that the tracing controller is no longer

used to write received tracing messages into the trace file" (pg 13, lines 9-12, cited

portion not included).

        Examiner respectfully disagrees. Shah disclosed: "informing the logging

controller that the logging controller is to write received messages into both the log file

and the trace file so that the tracing controller is no longer used to write received tracing

messages into the trace file ([0043], lines 1-5 and Fig. 7, where multiple loggers can

send their output to one handler, in Fig 7 message logger 703, or logging controller, and

trace logger 704, or tracing controller, both send output to file handler 706 which is in

contrast to Fig 5, where trace logger and message logger send their output to different

handlers)".

Conclusion

20.   Applicant's amendment necessitated the new ground(s) of rejection presented in
this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP
§ 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37
CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE
MONTHS from the mailing date of this action. In the event a first reply is filed within
TWO MONTHS of the mailing date of this final action and the advisory action is not
mailed until after the end of the THREE-MONTH shortened statutory period, then the
shortened statutory period will expire on the date the advisory action is mailed, and any
extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of
the advisory action. In no event, however, will the statutory period for reply expire later
than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the
examiner should be directed to MATTHEW S. LINDSEY whose telephone number is
(571)270-3811. The examiner can normally be reached on Mon-Thurs 7-5, Fridays 7-
12.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's
supervisor, John Follansbee can be reached on (571) 272-3964. The fax phone

number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

MSL
7/1/2009

/John Follansbee/
Supervisory Patent Examiner, Art Unit 2451